

Sistemas de Gestión de Bases de Datos: Configuración de una Instancia en Multipass®

Juannie

08/08/2024

El Script

Este script en Bash automatiza la creación y configuración de una máquina virtual (MV) usando **multipass**, una herramienta de Canonical para gestionar máquinas virtuales ligeras. Aquí te explico detalladamente cada parte del script:

```
#!/bin/bash

#####
##
↳ ##
## ¡Importante! Este script supone que Multipass está actualmente instalado en su
↳ sistema.  ##
↳
##
↳ ##
## El propósito del mismo es crear una máquina virtual (MV) con los parámetros de
↳ nombre, - ##
## cantidad de procesadores, tamaño de disco y cantidad de memoria RAM deseados.
↳ ##
##
↳ ##
## Una vez configurada la MV en Multipass, se inicia una instancia y se instala en ella
↳ - ##
## algunos paquetes que necesitaremos para trabajar en la cátedra.
↳ ##
##
↳ ##
## Este script funciona tanto para GNU/Linux como para Mac y Windows.
↳ ##
##
↳ ##
#####

nombremv="mv-$(RANDOM)"

cpu=2

disco=30G
```

```

memoria=3G

clear
date

echo "Creando máquina virtual '$nombremv' con:
- Procesadores: $cpu
- Tamaño de disco: $disco
- Memoria RAM: $memoria
..."
multipass launch -n $nombremv -c $cpu -d $disco -m $memoria 22.04
multipass mount $HOME $nombremv:Home
multipass set client.primary-name=$nombremv
multipass info $nombremv

echo "Máquina virtual creada. Preparando para su uso..."
multipass exec -n $nombremv -- sudo apt update
multipass exec -n $nombremv -- sudo apt -y upgrade
multipass restart $nombremv

date

multipass exec -n $nombremv -- sudo apt install -y avahi-daemon winbind language-pack-es
↳ manpages-es

multipass exec -n $nombremv -- sudo update-locale LANG=es_AR.UTF-8

```

Explicación de su funcionamiento

1. Shebang y Comentarios

```

#!/bin/bash
#####
##                                     ##
↳
## ¡Importante! Este script supone que Multipass está actualmente - ##
## - instalado en su sistema.                                     ##
↳
##                                     ##
↳
## ...                                                         ##
##                                                         ##
#####

```

- `#!/bin/bash`: Indica que el script debe ejecutarse usando el intérprete de comandos Bash.
- Los comentarios explican que el script creará una máquina virtual con especificaciones de CPU, disco y RAM dadas.

2. Definición de Variables

```
nombremv="mv-$(RANDOM)"
cpu=2
disco=30G
memoria=3G
```

- `nombremv="mv-$(RANDOM)"`: Crea un nombre aleatorio para la máquina virtual, prefijado con “mv-” seguido de un número generado por `RANDOM`, que es una variable interna de Bash que produce un número entero aleatorio.
- `cpu=2`: Define la cantidad de núcleos de CPU a asignar (2).
- `disco=30G`: Define el tamaño del disco virtual (30 GB).
- `memoria=3G`: Define la cantidad de memoria RAM a asignar (3 GB).

3. Limpieza de Pantalla y Fecha/Hora

```
clear
date
```

- `clear`: Limpia la pantalla del terminal.
- `date`: Muestra la fecha y hora actuales, para tener un registro del momento en que se ejecuta el script.

4. Creación de la Máquina Virtual

```
echo "Creando máquina virtual '$nombremv' con:
- Procesadores: $cpu
- Tamaño de disco: $disco
- Memoria RAM: $memoria
..."
multipass launch -n $nombremv -c $cpu -d $disco -m $memoria 22.04
```

- `echo`: Imprime un mensaje indicando las especificaciones con las que se creará la máquina virtual.
- `multipass launch -n $nombremv -c $cpu -d $disco -m $memoria 22.04`: Lanza una nueva máquina virtual con:
 - `-n $nombremv`: Nombre de la máquina virtual.
 - `-c $cpu`: Cantidad de núcleos de CPU.
 - `-d $disco`: Tamaño del disco.
 - `-m $memoria`: Cantidad de RAM.
 - `22.04`: Especifica la versión de Ubuntu a usar (22.04 LTS).

5. Montaje de Directorio y Configuración de Multipass

```
multipass mount $HOME $nombremv:Home
multipass set client.primary-name=$nombremv
multipass info $nombremv
```

- `multipass mount $HOME $nombremv:Home`: Monta el directorio `HOME` del host en la máquina virtual bajo el directorio `Home`.

Más específicamente, este comando hace lo siguiente:

1. **Montar un Directorio:** `multipass mount` es un comando que permite compartir un directorio del sistema operativo host (tu computadora) con una máquina virtual creada con `multipass`.
2. **Directorio del Host:** La variable `$HOME` representa el directorio personal del usuario en el sistema operativo host (por ejemplo, `/home/tu-usuario` en Linux o macOS). Esto incluye todos tus archivos personales, configuraciones y otros datos almacenados en tu cuenta de usuario.
3. **Destino en la Máquina Virtual:** `$nombremv:Home` indica el punto de montaje dentro de la máquina virtual. En este caso, se monta en un directorio llamado `Home` dentro de la máquina virtual.

Significado Exacto: Este comando esencialmente hace que el directorio personal de tu usuario en el host esté accesible desde la máquina virtual en la ubicación `Home`. Es decir, dentro de la máquina virtual, puedes acceder a todos los archivos y carpetas que están en tu directorio personal del host como si fueran parte de la máquina virtual, pero no se están copiando, sino que se están compartiendo.

Ejemplo: Si en tu máquina host tienes un archivo en `/home/tu-usuario/documentos/archivo.txt`, después de ejecutar este comando, en la máquina virtual podrás acceder a ese archivo usando la ruta `/Home/documentos/archivo.txt`.

Esto es útil para compartir datos entre el host y la máquina virtual sin necesidad de copiar archivos, facilitando el trabajo entre ambos entornos.

- `multipass set client.primary-name=$nombremv`: Establece la máquina virtual creada como la primaria en el cliente `multipass`.
- `multipass info $nombremv`: Muestra información detallada de la máquina virtual creada.

6. Preparación de la Máquina Virtual

```
echo "Máquina virtual creada. Preparando para su uso..."
multipass exec -n $nombremv -- sudo apt update
multipass exec -n $nombremv -- sudo apt -y upgrade
multipass restart $nombremv
```

- `echo`: Informa que la máquina virtual ha sido creada y está siendo preparada.
- `multipass exec -n $nombremv -- sudo apt update`: Ejecuta un `apt update` dentro de la máquina virtual para actualizar la lista de paquetes disponibles.
- `multipass exec -n $nombremv -- sudo apt -y upgrade`: Ejecuta un `apt upgrade` para instalar las actualizaciones disponibles.
- `multipass restart $nombremv`: Reinicia la máquina virtual para que todos los cambios surtan efecto.

7. Instalación de Paquetes Adicionales

```
date
multipass exec -n $nombremv -- sudo apt install -y avahi-daemon winbind language-pack-es
↪ manpages-es
multipass exec -n $nombremv -- sudo update-locale LANG=es_AR.UTF-8
```

- `date`: Muestra nuevamente la fecha y hora, marcando el progreso del script.
- `multipass exec -n $nombremv -- sudo apt install -y avahi-daemon winbind language-pack-es manpages-es`: Instala los siguientes paquetes en la máquina virtual:
 - `avahi-daemon`: Para soporte de descubrimiento de red.

- `winbind`: Para integración con dominios de Windows.
- `language-pack-es`: Paquete de idioma español.
- `manpages-es`: Páginas de manual en español.
- `multipass exec -n $nombremv -- sudo update-locale LANG=es_AR.UTF-8`: Actualiza la configuración regional de la máquina virtual para usar el español de Argentina.

Resumen

Este script automatiza la creación de una máquina virtual con `multipass`, configurando su CPU, disco, y RAM, instalando actualizaciones, configurando montajes, e instalando paquetes adicionales. Es útil para quienes necesitan un entorno virtual estandarizado rápidamente, con configuraciones específicas y soporte para idioma español.